ONLINE QUIZ SYSTEM

_____

A Thesis

Presented to the

Faculty of

San Diego State University

_____

In Partial Fulfillment

of the Requirements for the Degree

Master of Science

in

Computer Science

_____

by

Prasanthi Kornepati

Spring 2017

SAN DIEGO STATE UNIVERSITY

The Undersigned Faculty Committee Approves the

Thesis of Prasanthi Kornepati:

Online Quiz System

Carl Eckberg, Chair
Department of Computer Science

William Root
Department of Computer Science

J. Carmelo Interlando
Department of Mathematics & Statistics

11/21/2016
Approval Date

DEDICATION

I dedicate my dissertation work to my beloved husband Satya Venkata Sudheer Yarrarapu and my son Sushanth Sai Yarrarapu. A special feeling of gratitude to my loving parents and my In-laws, whose words of encouragement and push for tenacity ring in my ears.

ABSTRACT OF THE THESIS

Online Quiz System
by
Prasanthi Kornepati
Master of Science in Computer Science
San Diego State University, 2017

OnlineQuizSystem is an intelligent prototype developed as a multiple choice question examination system which is built as a Web based application scalable to run on both intranet and internet. OnlineQuizSystem is intelligent enough and equipped with a level of automation to guide the students to get proficiency/become subject matter experts by taking the tests.

The core underlying automation of the OnlineQuizSystem will generates questions and options on the fly basis and captures the answers from students into the database. It ensures that students will have the Mock Tests for practice, and can get the results displayed in their panel.

The OnlineQuizSystem is designed for three major roles. Those are

- ✓ An administrator who controls all the level of roles and creates the tests.
- ✓ An Instructor should be able to create test questions and manage tests.
- ✓ A Student who is an end user for the tests.

The main objective of the OnlineQuizSystem is to efficiently evaluate the student using a fully automated system that not only saves a lot of time but also gives fast results.

Technologies used to develop this website are HTML, CSS, and JavaScript for client side validation, C# & ASP.NET for dynamic content update. Microsoft SQL Server is used as a data base server for this application.

The system has been seeded with about 300 questions that would allow generation of multiple choice questions of tester selected topics from an undergraduate class in relational databases.

Grading can be done immediately, for student practice purposes, or if the exam is given simultaneously to student of a class, grading can be delayed until everyone is done with the quiz.

Another salient features of Online Quiz system is the support for templated question formats. The main advantage of templated questions are, during an assessment each student will receive different questions based on same concept. This helps faculty to avoid any malpractices.

TABLE OF CONTENTS

PAGE

LIST OF TABLES

PAGE

LIST OF FIGURES

PAGE

ACKNOWLEDGEMENTS

I wish to thank my committee members who were more than generous with their expertise and precious time. Thank you Professor William Root, and J. Carmelo Interlando for agreeing to serve on my committee. I would like to acknowledge and thank Professor Carl Eckberg for allowing me to conduct my research and providing any assistance requested.

I would like to thank all the faculty members that assisted with beta testing of the project. Their excitement and willingness to provide feedback made the completion of this research an enjoyable experience.

I would also like to thank my parents, sister & brother. They were always supporting me and encouraging me with their best wishes.

Finally, I would like to thank my husband. He was always there cheering me up and stood by me through the good times and bad.

CHAPTER 1

INTRODUCTION

On-line Exams are very popular and useful for many Educational Institutes to prepare an exam. It is beneficial for both students and institutes because of the flexibility facilitated with respect to time and place. This site is an attempt to provide an acceptable environment for both Test Conductors and Students appearing for Examination.

Purpose of Online exams is to provide all the requirements which an Examination system must have to conduct online exams, with the user-friendly interfaces. Students can take this exams from home or work place or some other places at his/her convenience without the need of going to a particular physical destination and can view the results at the same time. Thus the goal of the site is to develop a system that saves the efforts and time of both the institutes and the students.

Online Quiz System is a web site that creates a network between the instructors and the students. Instructors create a test with the questions they want in the exam. Students are the end users to take these tests. The answers entered by the students gets evaluated and their score is calculated and displayed at the end of the test. This score can be used by instructors to evaluate the performance of the students. Online Quiz System provides the platform but does not directly involved in conducting the tests. Questions are posted by the users of the site. This system has been created with a strong and secured data base that provides accessibility to the administrators who have a valid user id and password. Administrator keeps an eye on overall functioning of the system.

The efficient use of "On-line Quiz System", any Educational Institute or Academy or Universities can use it to optimize their strategy for organizing the exams, and for getting better results in less time.

The system has been initialized and tested with roughly 300 questions appropriate for an undergraduate class on databases. It consists of theoretical multiple choice quiz as well as

templated questions. The questions are grouped together based on Topics. Online Quiz System consists of the following topics and has the provision to create new topics and add new questions.

1. The Relational Model.
2. SQL: Queries.
3. SQL: Constraints.
4. SQL: Triggers.
5. SQL: Nested Queries.
6. Relational Algebra.
7. Database Systems Design and Implementation.
8. Normalization.
9. ORACLE –JDBC.
10. Storage and File Structures.
11. Indexing and Hashing.
12. Query Processing Techniques.

Another salient feature of the Online Quiz system is the support for templated question formats. The main advantage of templated questions are, during an assessment each student will receive different a question based on the same concept. This helps faculty to conduct quizzes simultaneously for many student and also avoid malpractices.

Quite a few years ago, a similar tool was built for databases, and was used for quizzes for CS-514 class on databases. That tool has since gone missing, and in any case, this version is a technological upgrade, designed in such a way as to support any class, or subject matter.

This document describes how this application was put together, from its requirements, to the tools and technologies used to build this application. It is divided into 6

Chapters. Chapter 2 discusses the technologies used to create this tool. Chapter 3 presents an overview of the Software Design to implement this product. Chapter-4 focusses on implementation part of this application. Chapter 5 gives us some closing points in the form of conclusion and discusses the obstacles faced in implementing this application and finally, Chapter 6 talks about the probable future enhancements and improvements to the existing application.

CHAPTER 2

TECHNOLOGIES

Several technological tools were used in this project; each is summarized in the sections that follow.

2.1 HTML

Hyper Text Markup Language (HTML) is the standard markup language for creating web pages and web applications. Along with CSS, and JavaScript, HTML forms a cornerstone technology used to create web pages, [1] as well as to create user interfaces for mobile and web applications. Web browsers can read HTML files and render them into visible or audible web pages. HTML describes the structure of a website semantically and, before the advent of Cascading Style Sheets (CSS), included cues for the presentation or appearance of the document (web page), making it a markup language, rather than a programming language.

HTML elements form the building blocks of HTML pages. HTML allows images and other objects to be embedded and it can be used to create interactive forms. It provides a means to create structured documents by denoting structural semantics for text such as headings, paragraphs, lists, links, quotes and other items. HTML elements are delineated by tags, written using angle brackets. Tags such as <img /> and <input /> introduce content into the page directly. Others such as ... surround and provide information about document text and may include other tags as sub-elements. Browsers do not display the HTML tags, but use them to interpret the content of the page.

2.1.1 HTML5 Pros

- Arguably, HTML5 apps do perform relatively well, and also have the added advantage that they are portable.
- Also, web-rendering has more sophisticated rendering primitives that offer a lot of flexibility and control to the programmer [2].
- Text-layout is very sophisticated and automatic.

2.1.2 HTML5 Cons

- The main problem with HTML5's acceptance is that only modern browsers support it.
- Another problem is that although parts of the language are very stable, the language itself is considered a work in progress, so technically, any of the elements could change at any time.
- The language is not expected to be completed for several years, which complicates things further. Thankfully, a lot of the language is considered stable and ready to use.
- I think it's such a great move forward, that you should develop a Graceful Degradation approach to writing your HTML. That simply means writing HTML that will work with older browsers, but will offer users with more modern browsers an enhanced experience.

2.2 CSS

Cascading Style Sheets (CSS) is a style sheet language used for describing the presentation of a document written in a markup language [3]. Although most often used to set the visual style of web pages and user interfaces written in HTML and XHTML, the language can be applied to any XML document, including plain XML, SVG and XUL, and is applicable to rendering in speech, or on other media. Along with HTML and JavaScript, CSS is a cornerstone technology used by most websites to create visually engaging webpages, user interfaces for web applications, and user interfaces for many mobile applications.

CSS is designed primarily to enable the separation of document content from document presentation, including aspects such as the layout, colors, and fonts [3]. This separation can improve content accessibility, provide more flexibility and control in the specification of presentation characteristics, enable multiple HTML pages to share formatting by specifying the relevant CSS in a separate .css file, and reduce complexity and repetition in the structural content.

Styles define how to display HTML elements. CSS overrides the browser's default settings for interpreting how tags should be displayed, letting you use any HTML element indicated by an opening and closing tag to apply style attributes defined either locally or in a style sheet.

External Style Sheets can save a lot of work. They are stored in CSS files. By separating visual design elements (fonts, colors, margins, and so on) from the structural logic of a web page, CSS gives designers the control they crave without sacrificing the integrity of the data, thus maintaining its usability in multiple environments.

2.2.1 CSS Pros

- Speed of Design: CSS helps developers quickly create multiple pages in the same style and format.
- Consistency: CSS helps establish a consistent framework that Web designers can use across all the sites they build [4].
- Ease of Use: CSS is easy to learn, and there are many tutorials and forums online to help designers in a pinch.
- Multiple Browser Support: CSS works with Internet Explorer, Firefox, Chrome and more.

2.2.2 CSS Cons

- Speed: Downloading an HTML page will always take longer if CSS is embedded within it. However, with ever-increasing Internet speeds, this is less of a problem than you might imagine.
- Weirdness: While CSS is easy to use and understand, its syntax is very different from HTML and not terribly user-friendly. This forces developers to take time to learn two different types of code and then understand how they interact with each other.
- Complications: CSS can get messy and complicate the creation of websites with third-party software, such as Dreamweaver or Microsoft FrontPage.

2.3 JAVASCRIPT

Javascript is a dynamic computer programming language [5]. It is lightweight and most commonly used as a part of web pages, whose implementations allow client-side script to interact with the user and make dynamic pages. It is an interpreted programming language with object-oriented capabilities.

JavaScript was first known as LiveScript, but Netscape changed its name to JavaScript, possibly because of the excitement being generated by Java. JavaScript made its first appearance in Netscape 2.0 in 1995 with the name LiveScript. The general-purpose core of the language has been embedded in Netscape, Internet Explorer, and other web browsers.

Client-side JavaScript is the most common form of the language. The script should be included in or referenced by an HTML document for the code to be interpreted by the browser. It means that a web page need not be a static HTML, but can include programs that interact with the user, control the browser, and dynamically create HTML content.

The JavaScript client-side mechanism provides many advantages over traditional CGI server-side scripts. For example, you might use JavaScript to check if the user has entered a valid e-mail address in a form field. The JavaScript code is executed when the user submits the form, and only if all the entries are valid, they would be submitted to the Web Server.

2.3.1 JavaScript Pros

- Wide browser support i.e. Most Browsers speak a dialect.
- Easy access to document objects and their manipulation
- No long download times w.r.t java or graphic animations
- No Plug-in support required
- Relatively Secure [6]

2.3.2 JavaScript Cons

- Not standard support for JavaScript across browsers esp. DOM
- Web page useless if script does not work!!
- JavaScript may be disabled by browser reading HTML file no control over this
- JavaScript can run slowly

2.4 C-SHARP

C#(pronounced as see sharp) is a multi-paradigm programming language encompassing strong typing, imperative, declarative, functional, generic, object-oriented (class-based), and component-oriented programming disciplines [7]. It was developed by Microsoft within its .NET initiative and later approved as a standard by Ecma (ECMA-334)

and ISO (ISO/IEC 23270:2006). C# is one of the programming languages designed for the Common Language Infrastructure.

2.4.1 C# Pros

- For many, C# has a familiar syntax in that it follows C.
- Fully integrated with the .NET library, providing access to a great repository of functionality and support [8]
- Safer than its namesakes (C/++) in that pointer types are not permitted.

2.4.2 C# Cons

- Obviously less efficient in execution than C/++ because it is ultimately interpreted.
- Still allows pointers in 'unsafe' blocks.
- Loses any cross-platform capabilities because of the integration with .NET.

2.5 ASP.NET

ASP.NET is an open-source [9] server-side web application framework designed for web development to produce dynamic web pages. It was developed by Microsoft to allow programmers to build dynamic web sites, web applications and web services.

It was first released in January 2002 with version 1.0 of the .NET Framework, and is the successor to Microsoft's Active Server Pages (ASP) technology. ASP.NET is built on the Common Language Runtime (CLR), allowing programmers to write ASP.NET code using any supported .NET language. The ASP.NET SOAP extension framework allows ASP.NET components to process SOAP messages.

ASP.NET is in the process of being re-implemented as a modern and modular web framework, together with other frameworks like Entity Framework. The new framework will make use of the new open-source .NET Compiler Platform (code-name "Roslyn") and be cross platform. ASP.NET MVC, ASP.NET Web API, and ASP.NET Web Pages (a platform using only Razor pages) will merge into a unified MVC 6, where MVC denotes the Model View Controller design pattern.

2.5.1 ASP.NET Pros

- Quick drag-and-drop controls.
- Pseudo-hides the web page cycle from developers who are not web-friendly.
- C# is a much nicer language than PHP. VB.NET is perhaps too.
- Validation and AJAX are pretty easy.

2.5.2 ASP.NET Cons

- If you know the web then ASP.NET can get in the way.
- The cost if you move to an enterprise level can be larger, but for the most part there are various legitimate ways to get discounted or often free software under Microsoft programs.
- Harder to find cheap labor.
- Often obscure bugs related to leaky abstractions.
- If you have developers who know what they're doing on the web, and are decent coders too (a programming background) then ASP.NET MVC is the way to go.

2.5.3 ASP.NET MVC Pros

- Gets out of your way / minimalist approaches to code and markup.
- Minimal bugs and typically easy to correct the ones that exist.
- Highly testable [10].
- Highly modular (can write views as PHP style code if you wanted, or HAML or anything).

2.5.4 ASP.NET MVC Cons

- No drag-and-drop controls (same as PHP).
- The cost is the same as ASP.NET.
- Harder to find coders Combined Client+Server validation still feels like it's getting there, but it's on par with PHP at least and there's a lot of cool auto-binding

2.5.5 ASP.NET vs PHP

While there are many programming languages out there for a web developer to choose from, [11] the languages market is generally dominated by two competing language giants: ASP.NET and PHP. Both are good systems on their accord. Apparently Asp.Net has

the advantage of being able to utilize a more mature platform, [10] and better languages (but its personal taste), and PHP has more world ready open source applications. But these does not answer which one to use in a project.

It depends on many factors. For example:

- Size of the project (PHP might be better for small size quick solutions)
- Size of any existing codebase
- Requirements of the client.
- Deployment environment (Asp.Net is not good on Linux)
- Risks (Asp.Net comes with very good anti injection mechanisms, for PHP you'd need consistent use PEAR libraries)

Baseline: It's generally a business decision, not a technical one.

2.6 MICROSOFT SQL SERVER

Microsoft SQL Server [12] is a relational database management system developed by Microsoft. As a database server, it is a software product with the primary function of storing and retrieving data as requested by other software applications—which may run either on the same computer or on another computer across a network (including the Internet).

CHAPTER 3

DESIGN

3.1 OVERVIEW

The basic idea of the Online Quiz System (OQS) is to provide a web based solution for student assessment management. This portal is highly scalable and customizable for faculties of various departments in an institution to conduct online examinations and generate assessment reports. Students can benefit by taking assessments remotely as well as gaining in-depth knowledge of the course through mock tests.

Using this tool faculty members can create new assessments with just a few clicks and conduct examinations in a secure manner. Each assessment can be created from various topics of the course, and each topic contains a group of questions. The faculty has a provision to add/edit/delete a topic and add/update questions to the topics. They can also create mock tests, so that students can practice and become proficient in that domain.

Students assessment reports are private and can only be accessed by the faculty member or the authorized person. Faculty members should be able to view summarized assessment reports as well as a detail report for each student.

3.2 UI DESIGN

3.2.1 Login

The Online Quiz System is a secured web application and can be only accessed by an authorized faculty member, instructor and registered students. The system support 3 kinds of login roles.

3.2.1.1 ADMINISTRATOR LOGIN

This login is used by a super user. Administrator Login credentials are preconfigured in the database and provides a super user access rights to the website.

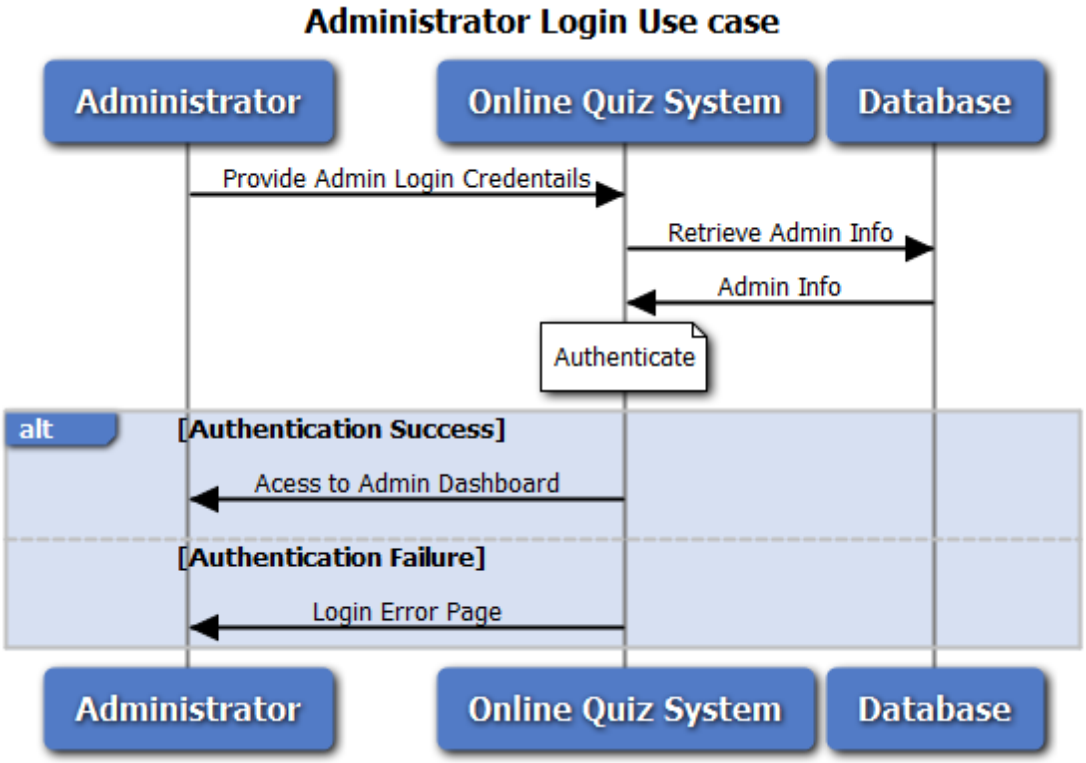

Figure 3.1. Administrator Login Use Case.

3.2.1.2 INSTRUCTOR LOGIN

Instructors need to register on the website from the Instructor Login page for initial access. Once an instructor is successfully registered on the website, the Administrator will be notified about a new instructor. The Administrator needs to authorize the instructor so that the instructor will get access to the Dash board.

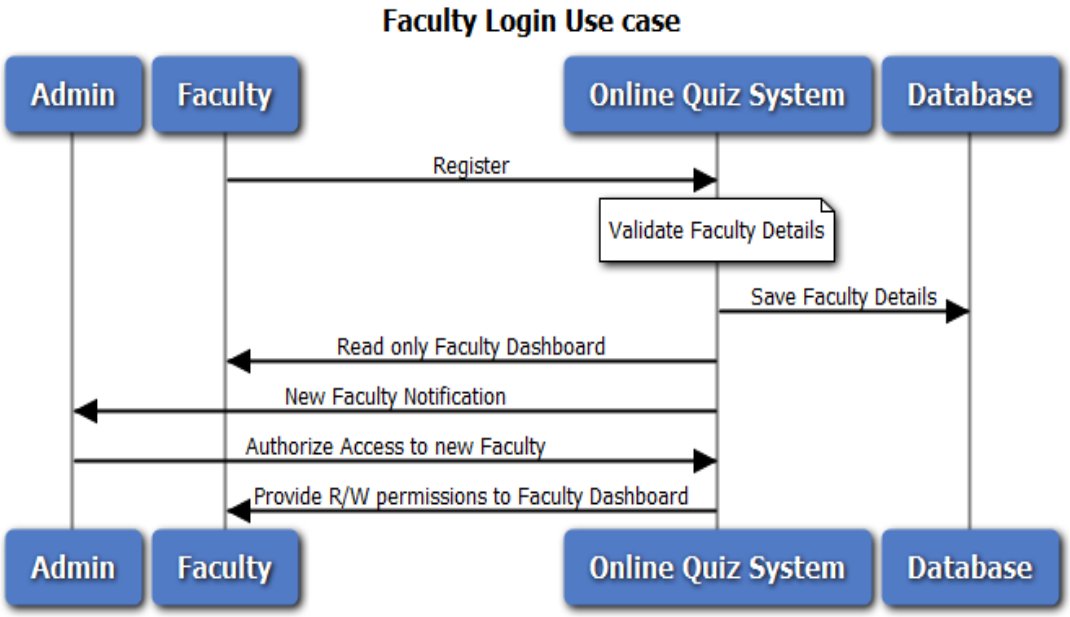

Figure 3.2. Faculty Login Use Case.

3.2.1.3 STUDENT LOGIN

Students need to register on the website by providing student details. Once a student creates his profile on the website, subsequently he can login using his credentials.

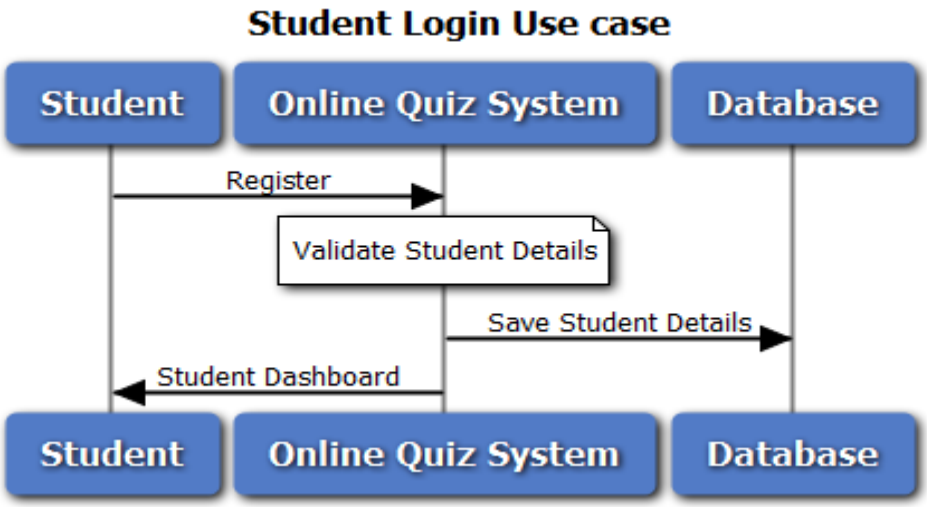

Figure 3.3. Student Login Use Case.

3.2.2 Administrator Dashboard

The Administrator Dash Board consists of the following menu items

3.2.2.1 TOPICS MENU

The Topics Menu lists the available Topics for the Quiz. The Administrator or an authorized faculty member can add a new topic, and update or delete an existing topic.

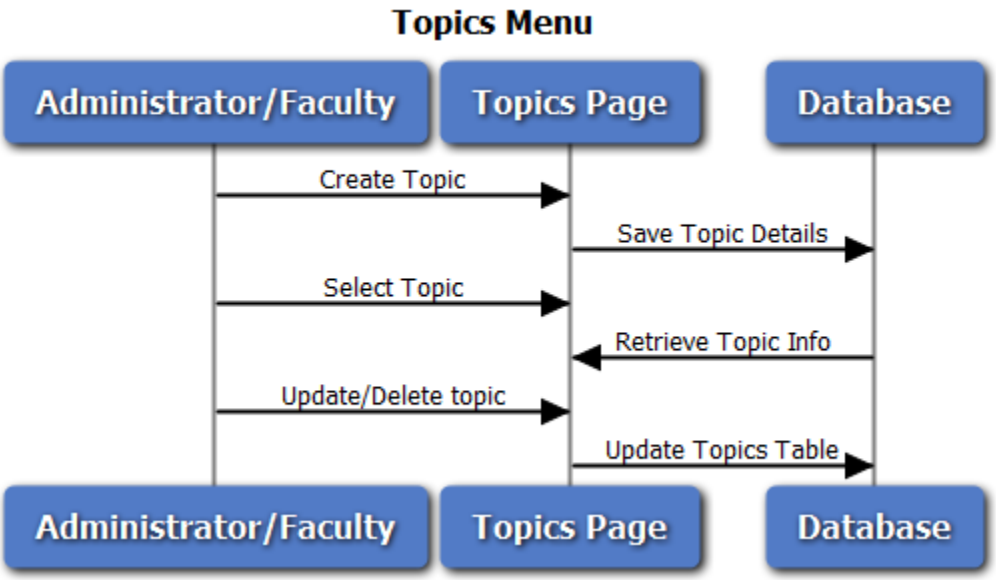

Figure 3.4. Topics Menu.

3.2.2.2 QUESTIONS MENU

In this Menu one can manage the available quiz questions in the data base. Administrator/Faculty can add new questions to a topic or delete/update the existing questions.

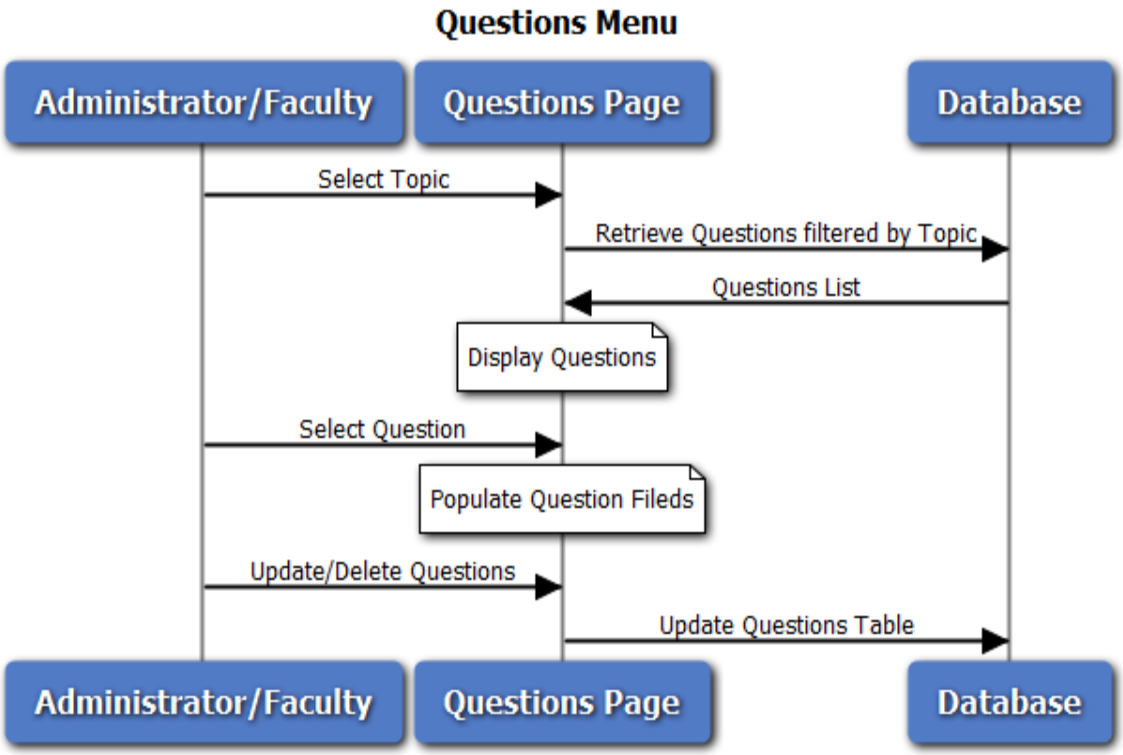

Figure 3.5. Questions Menu.

3.2.2.3 MANAGE ROLES

This menu helps the Administrator to manage faculty members. Through this menu an administrator can enable or disable access permission to existing faculty members. This menu will be available only to Administrator.

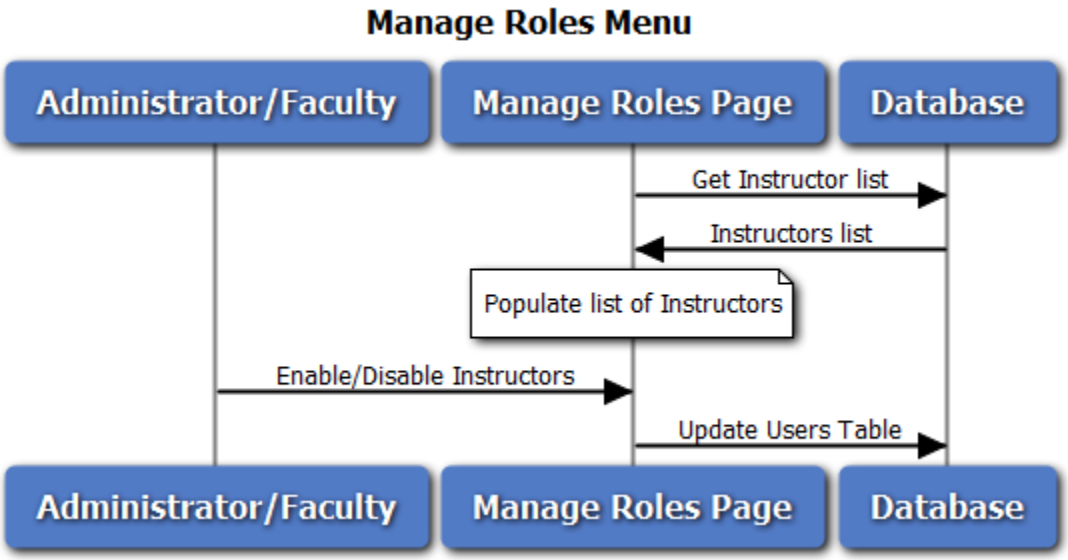

Figure 3.6. Manage Roles Menu.

3.2.2.4 APPROVE NEW INSTRUCTORS

The Administrator can use this menu to authorize new faculty members to the system. When new faculty register themselves on the Online Quiz System, a notification will pop-up on the Administrator Dash board. Admins can authorize new faculty from this page. This menu will be accessible only to an Administrator.

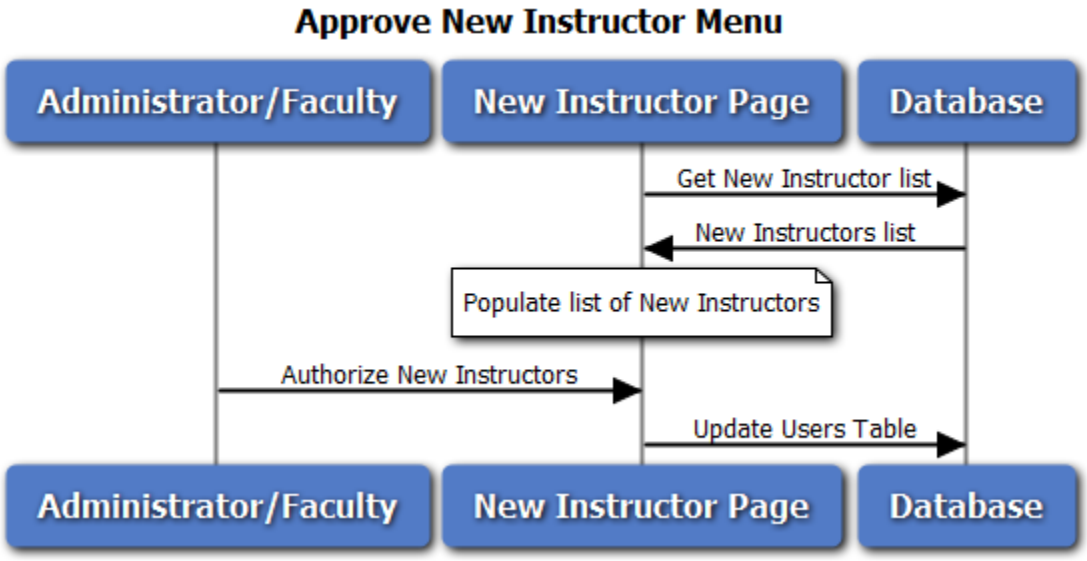

Figure 3.7. Approve New Instructor Menu.

3.2.2.5 ASSESSMENT REPORTS

All the assessment reports can be viewed once the administrator/faculty clicks on this menu.

Faculty can filter down the reports based on assessments. Once the assessment is selected from the drop down list, one can see the list of students that took the assessment. They can also select a particular student to view how they performed on that particular assessment.

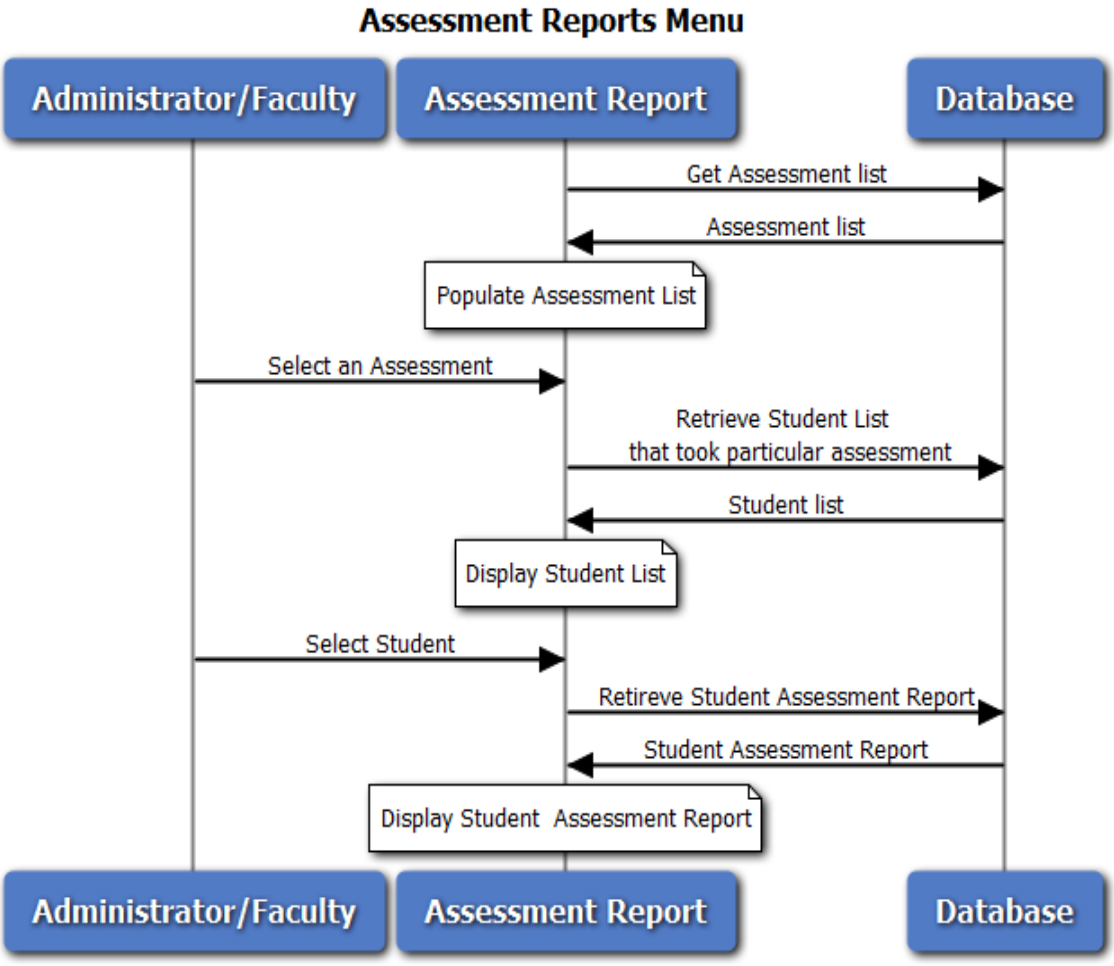

Figure 3.8. Assessment Report Menu.

3.2.2.6 TEST DASH BOARD

Test dashboard is used to create and configure the following.

1. Create an assessment.
2. Designate an assessment as a Real/Mock Test
3. Choose topics for the assessment

An assessment consists of questions from one or more topics. This Test Dashboard provides the flexibility for Admin/Faculty to configure the topics for an assessment. For example, if a faculty member wants to test students on 3 different topics, they can create an assessment that includes those 3 topics.

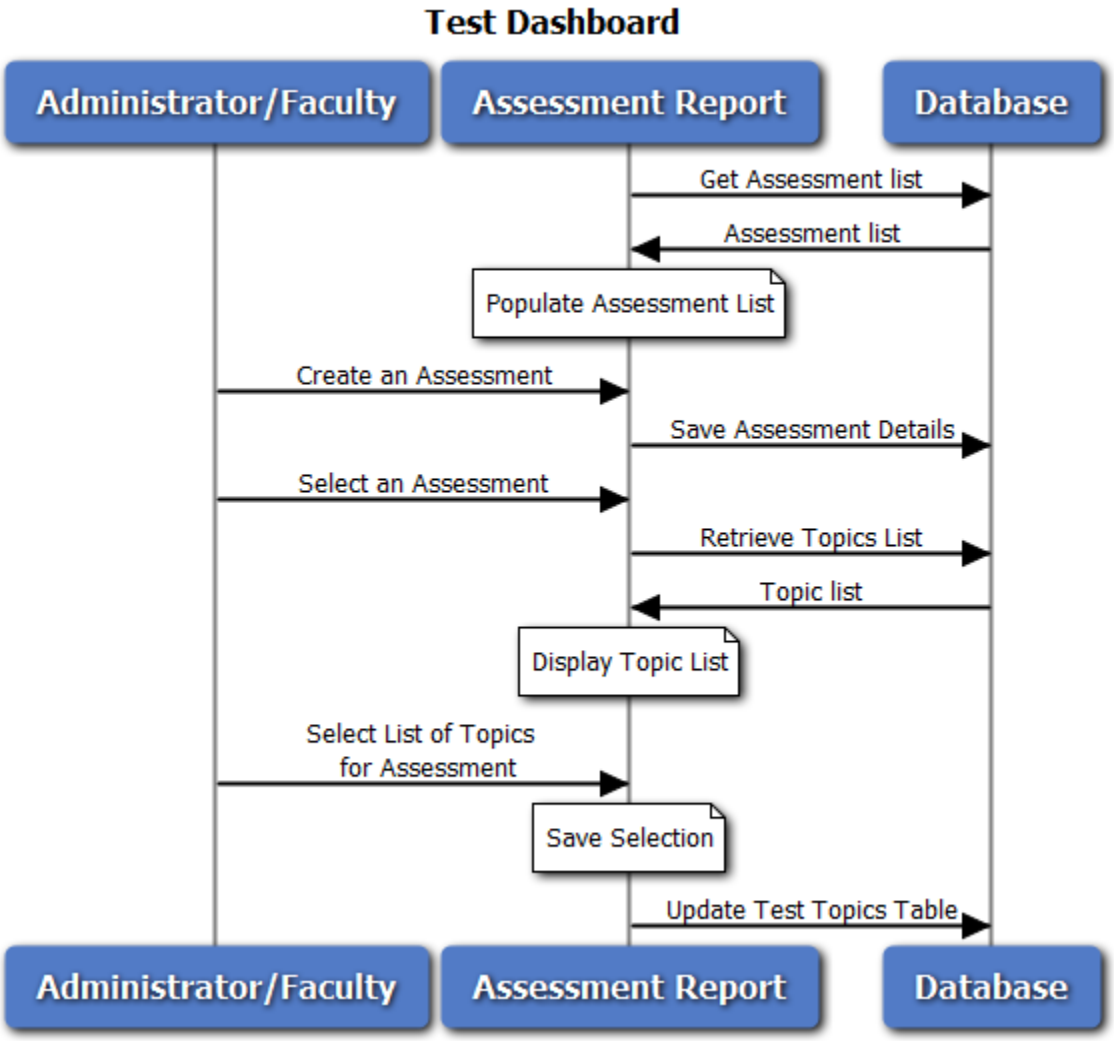

Figure 3.9. Test Dashboard.

3.2.2.7 USERS

The Users menu will display all the user lists that are registered in the Online Quiz System. They can be either Administrator or Faculty or Student. This menu is accessible only to the Administrator.

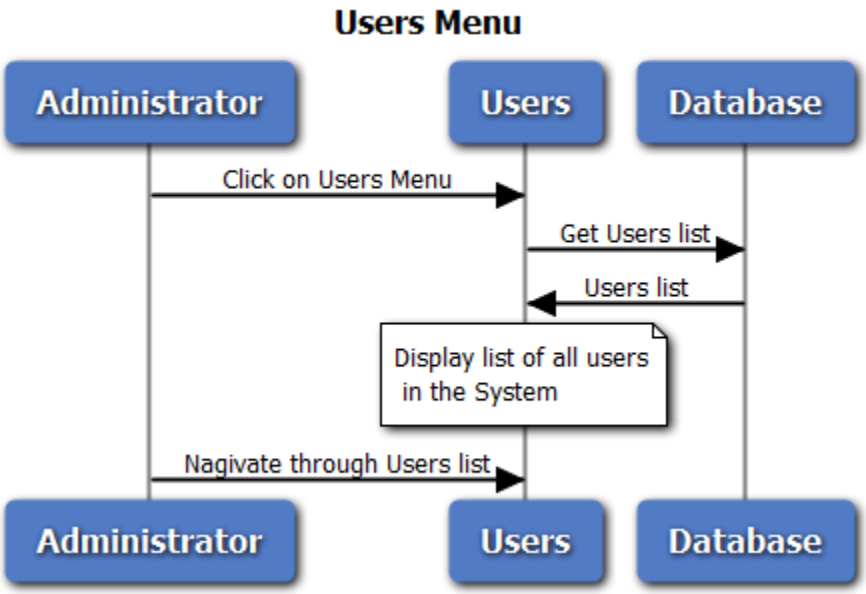

Figure 3.10. Users Menu.

3.2.3 Instructor Dashboard

The Instructor dash board is used by faculty. It is a subset of the Administrator Dashboard and consists of the following menu items which are described above.

- Topics
- Questions
- Assessment Reports.

3.2.4 Student Dashboard

The Student dashboard consists of the following menu items

- Self-Assessment
- History
- Mock Test

3.2.4.1 SELF-ASSESSMENT

This menu item is used by students to do self-assessments. Students can select an assessment from the drop down list and can take the quiz

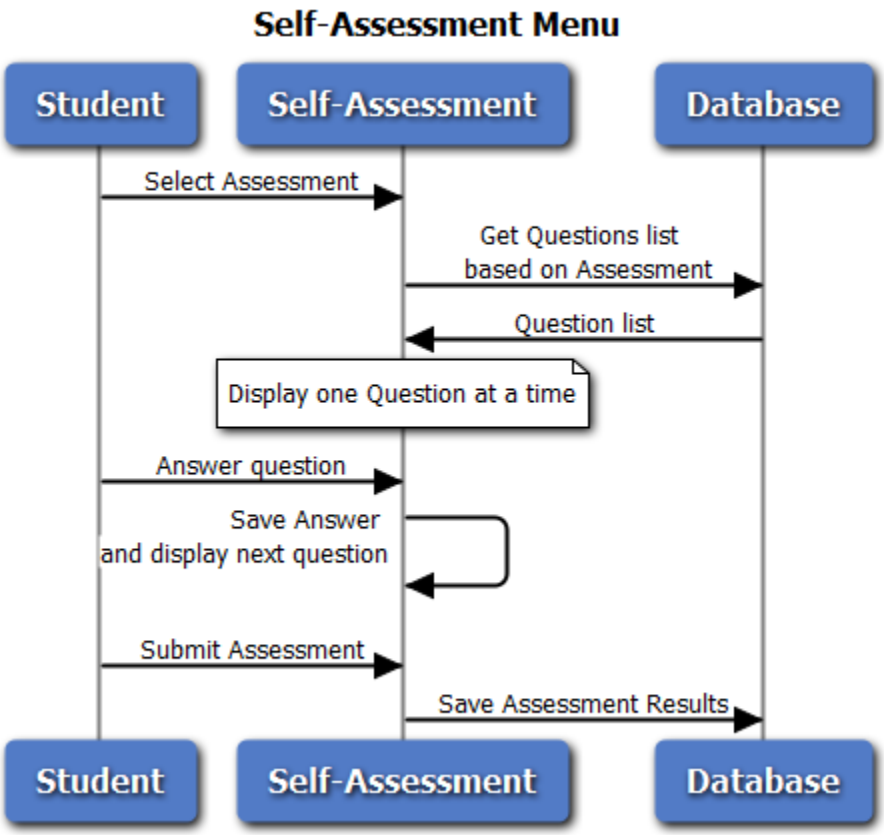

Figure 3.11. Self-Assessment Menu.

3.2.4.2 HISTORY

This menu displays a list of assessments that were taken by a student. It also provides information about test results as well as a summary of each test.

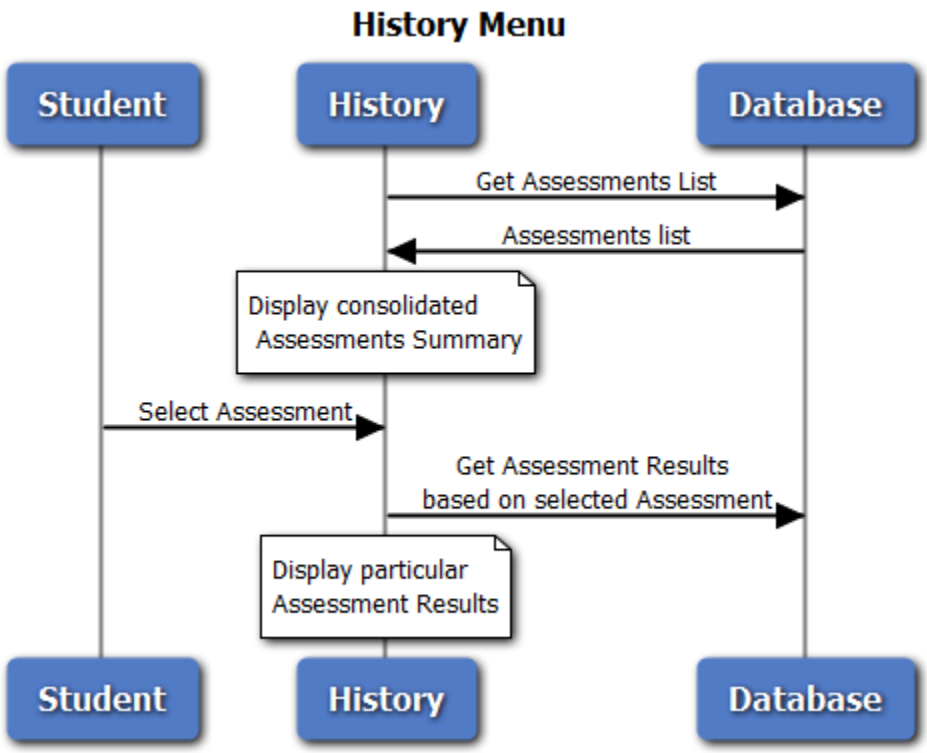

Figure 3.12. History Menu.

3.2.4.3 MOCK TEST

This menu is exactly the same as the Self-Assessment menu, except that the pool of questions are from a mock test. This menu is intended for students to self-evaluate their expertise in a particular domain as well as to practice for the self-assessment.

3.2.5 Logout

This screen is used by all the users to cleanly logout from the Online Quiz System.

3.3 DATA BASE DESIGN

The Online Quiz System (OQS) database has been designed in a sophisticated manner which uses stored procedures to perform database relational transactions. It consists of the following tables

3.3.1 Tables for User and Role Management

- Role Table : This table contains data about the roles in OQS
- User Information : Holds user information in OQS

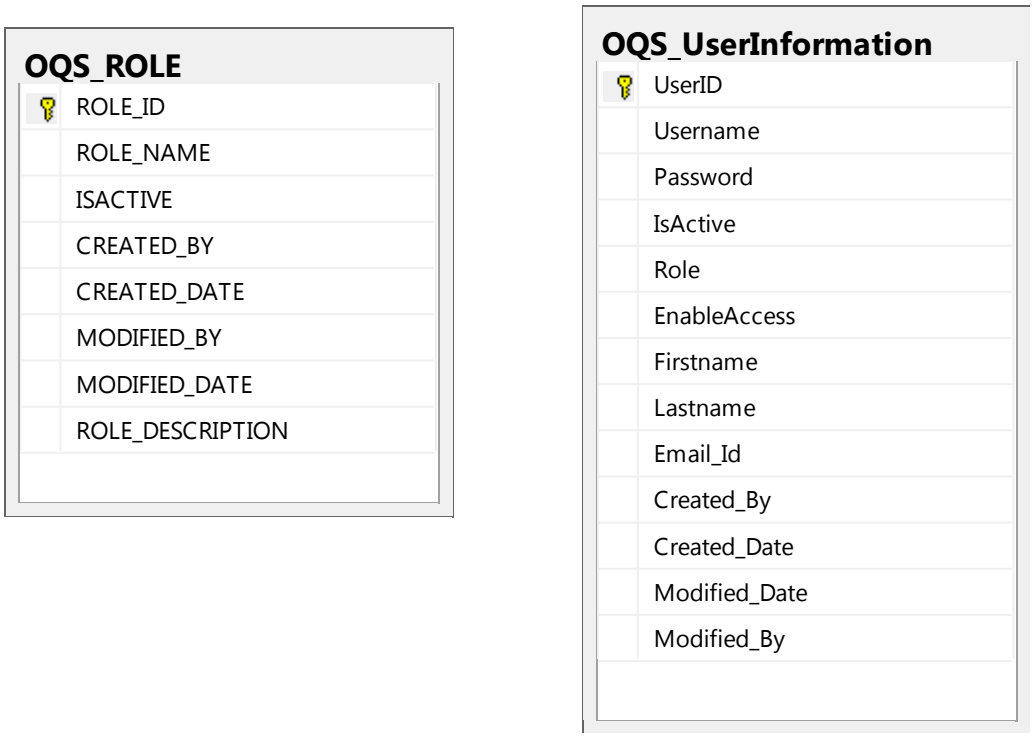

Figure 3.13. Role & User Information Table.

3.3.2 Tables for Questions and Assessment Management

- Questions Table : Contains all the questions available in the OQS
- Topics Table: Contains list of topics in OQS
- Tests Table : Contains list of tests or assessments in OQS
- Test Topics Table: Mapping table between Tests and Topics
- Answer Table : Contains records about the user selected answers

The relationship between tables is shown in Figure 3.14

- Topic_Id in Questions table references Topic_Id in Topics table
- Question_Id in Answers table references Question_Id in Question table
- TestId in TestTopics table references Id in Test table
- TopicId in TestTopics table references Topic_Id in Topics table.

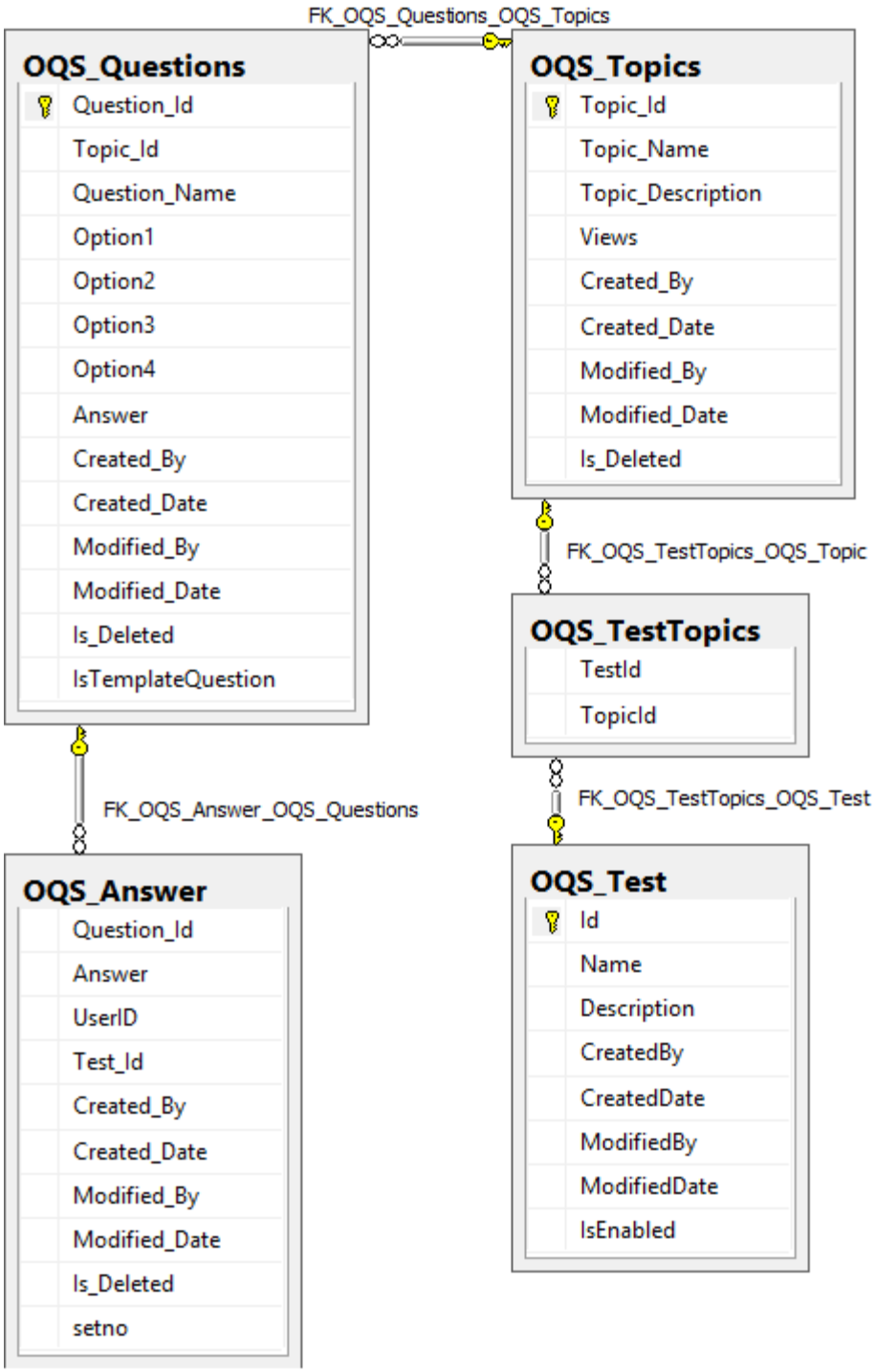

Figure 3.14. Tables & Relationships.

3.4 TEMPLATED QUESTIONS

One of the salient features of Online Quiz system is the support for templated question formats. The main advantage of templated questions are, during an assessment each student will receive different questions based on same concept [13, 14].

We have used database tables to design and implement templated question. Each entry in question table has a field to designate whether the question is a normal question or template question. If it is template question there will be a corresponding entries in TemplateQuestionSetInfo table that provides information on each set of values for placeholders in template questions. There is also a corresponding entry in TemplateQuestionOptionInfo table that provides information about the multiple choice answer options for each template set.

Figure 3.15 shows the tables that are used to implement templated questions.

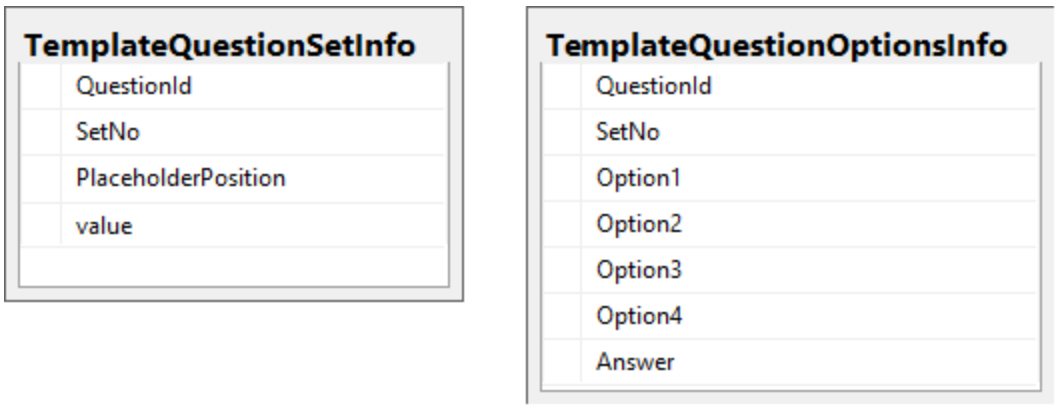

Figure 3.15. Tables for Templated Questions.

For example, the question below tests the relational algebra concept.

Suppose relation R (A, B) currently has tuples {0} and relation S (B, C) currently has {1}. What is the number of tuples in the result of the SQL query?

Select * From R Natural Join S;

In the above example {0} and {1} designates place holders in a templated question which are replaced from the following set of parameters. Thus Online Quiz System can generate questions randomly at runtime, such that each student observes a different question, but based on same concept

Table 3.1. Sample Data Sets for Templated Questions

Set#	Placeholder 1	Placeholder2	Answer
1	{(1,2), (1,3), (3,4)}	{(2,5), (4,6), (7,8)}	2
2	{(1,4), (1,3), (3,4)}	{(2,5), (4,6), (7,8)}	2
3	{(1,1), (1,3), (3,4)}	{(2,5), (4,6), (7,8)}	1

CHAPTER 4

IMPLEMENTATION

4.1 Overview

This section explains in detail the file structure and the coding process that was undertaken while designing and implementing the project.

4.2 Login Session

This section explains the files that have the relevant code to display the login page to handle the database connections for verification of the credentials entered by the user (See Figure 4.1).

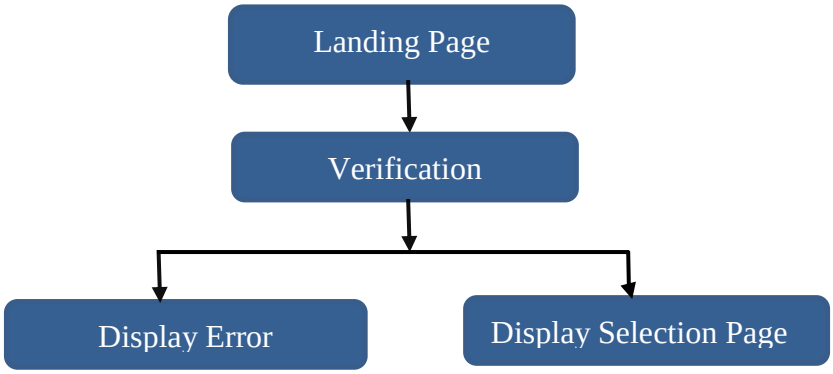

Figure 4.1. Overview of login session.

4.2.1 Landing Page

This secure login page is a gateway to the “Online Quiz System” web portal. The user is provided with a user-friendly login panel where he can input his login credentials that will be validated against the admin configured data. Every user must go through this login process in order to perform the actions relevant for his user roles. As soon as the user tries to browse any page in this application, he will be redirected immediately to this login page if he

was not authenticated earlier. Once the user is authenticated initially, his subsequent actions do not need to go through any authentication challenges again. Though HTTP is stateless we have used “ASP.NET” server side state management techniques (e.g. SESSIONS) to remember the user across the page post backs/requests.

We have implemented client side validations to save the number of round trips to the server during invalid submissions, “Asp.net validation controls” played a key role in implementing these client side validations.

The landing page is named “login.aspx” and contains the asp.net server controls code to display the webpage. It contains a form with two separate asp.net text boxes, one for entering the user ID and other for the password. It has two buttons, “Signup”, which allows creation of new users instantly and “Submit” which triggers the client side validation and eventually posts the user entered data. This request in turn will be validated with the admin configured data in the database. Figure 4.2 shows screenshot of the actual image of the login page.

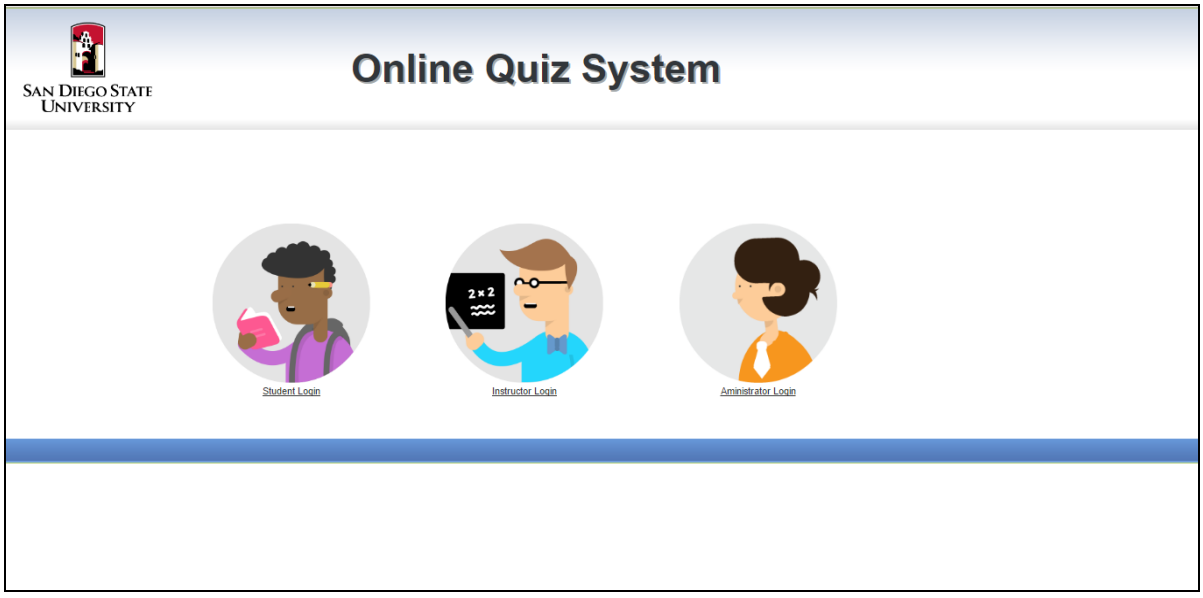

Figure 4.2. Landing Page.

4.2.2 Verification

Once the user enters the user name and password and clicks the “Submit” button, new request will be sent to the same login but as a post back request. And during postback requests, we make a connection to the database “OnlineQuizSystem” as shown in Figure 4.3.

```
public DataSet AuthenticateLogin(tLogin objtLogin)
{
    string strConn = ConfigurationSettings.AppSettings["dbUserModule"].ToString();
    SqlConnection sqlConn = new SqlConnection(strConn);
    sqlConn.Open();
    SqlCommand sqlCmd = new SqlCommand("usp_OQS_LoginQuiz", sqlConn);

    sqlCmd.CommandType = CommandType.StoredProcedure;
    sqlCmd.Parameters.AddWithValue("@UserName", objELogin.UserName);
    sqlCmd.CommandType = CommandType.StoredProcedure;
    sqlCmd.Parameters.AddWithValue("@Password", objELogin.Password);
    sqlCmd.CommandType = CommandType.StoredProcedure;
    sqlCmd.Parameters.AddWithValue("@Role_Name", objELogin.Role);
    SqlDataAdapter sqlDa = new SqlDataAdapter();
    sqlDa.SelectCommand = sqlCmd;
    DataSet dsLogin = new DataSet();
    sqlDa.Fill(dsLogin);
    return dsLogin;

}
```

Figure 4.3. Code snippet for database connection.

Once connected successfully, the SQL stored procedure script reads only those entries from the table “login” whose ID matches the one entered by the user. This table contains the list of all the valid user ids and the corresponding passwords The user id and the password is then compared with the one entered by the user and if they match, then the user is granted access and next page “home.aspx” is loaded If the entered credentials do not match with any entries from the database, the login page “login.aspx” with an error message is loaded.

4.3 HOME PAGE/ USER DASHBOARDS

Once a user is authenticated, he will be redirected to the customized dash board page based on his privileges recorded (role) in the system. We have primarily three kinds of roles in the “Online Quiz Portal” system.

1. Student Role
2. Administrator Role
3. Instructor Role

4.4 STUDENT DASHBOARD

The student home page is named as "Studentdefault.aspx" and contains well-formed asp.net server controls with externalized CSS styles and server side code to display a user-friendly webpage with options to view students past assessments history, initiate new assessments and self-assessments with mock tests. This implementation leverages CSS style sheets to give a consistent look and feel across the web application. Also, to give a better navigation experience to the portal, we have implemented "Asp.Net master page layout" concept to design the layout for student dashboard forms. Users will see all action links on the left panel of the web pages. These action links will remain the same irrespective of the action that the user is performing. However, the current user action will be highlighted in the bold blue color indicating that the user has been working on this specific action. This action links menu in the master page as shown in Figure 4.4.

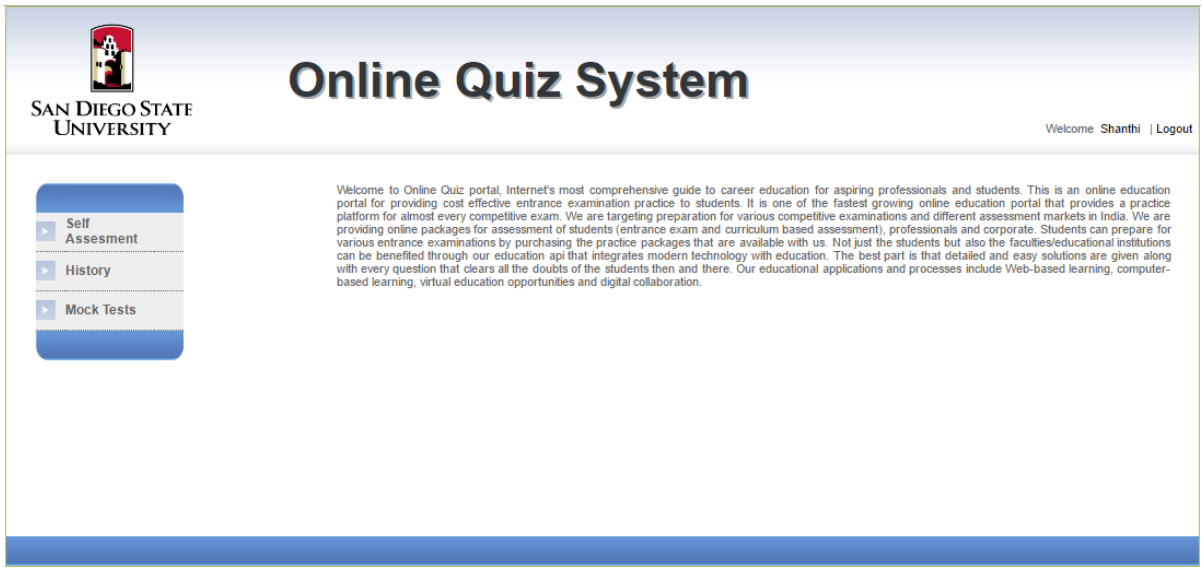

Figure 4.4. Action links in "Student Dashboard" page.

Students are the primary users of this application and have privileges to the core functionalities below.

1. Take assessment.
2. View assessments history.
3. Self-assessment with mock tests.

4.4.1 Take Assessment

Students take assessments configured by the instructor. They can choose a specific assessment out of dropdown list and can complete them in given time. Completed assessments will be shown under history action link on the next day of the assessment. This is implemented in “choosetest.aspx” page. When the user selects a specific test, we redirect him to “Questionaire.aspx”, which renders the list of all questions tagged under this specific test.

Assessment rendering is a complex operation. It requires showing one question at a time with easy navigation to the next and previous questions, and the user’s responses to the questions must be retained during the next and previous question clicks. Implementing this whole thing using the plain html or asp.net control is error prone and an inflexible design, so we have leveraged “ASP.NET GRID VIEW” control with pagination option to render the question layout as shown in Figure 4.5.

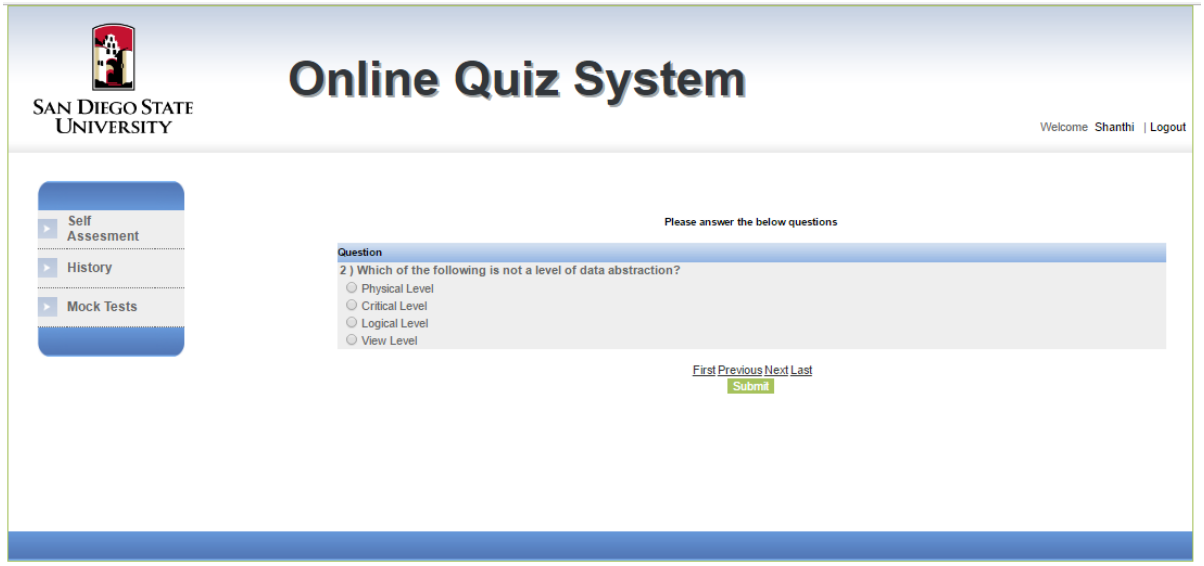

Figure 4.5. Assessment question layout using grid view with pagination.

While students traverse across the list of questions, their previous question responses were retained using a client side state management technique. When the user reviews his answers and submits the test, his responses are captured and saved in the “Answers” table in the database using stored procedures.

4.4.2 Self-Assessment with "Mock Tests"

This is another strong feature of this application, which helps students in improving their knowledge and performing well in the real assessments.

This is implemented as "MockQuestionaire.aspx" asp.net page, which is similar to rendering the assessment questionnaire, however we can immediately check the user answer (as soon as he chooses one of the answer options) and display a short message indicating the correctness of his answer. We also display the actual right answer as per the Instructor's input. This helps student to practice for their real assignments.

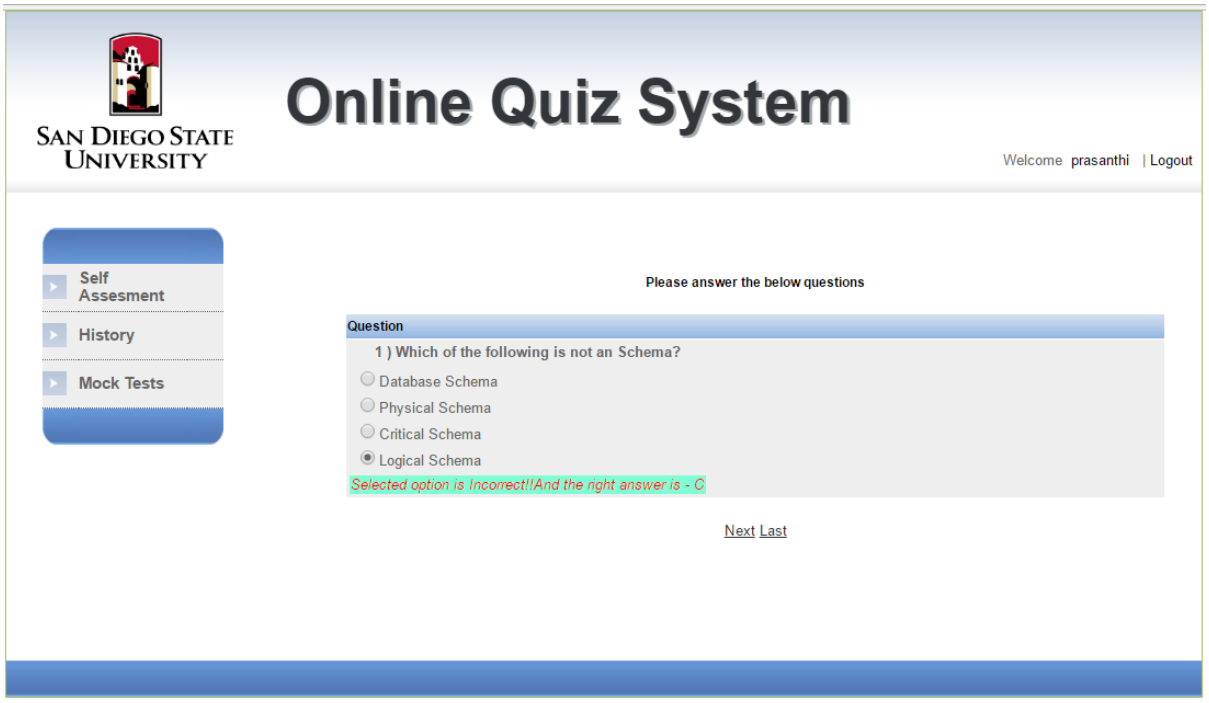

Figure 4.6. Self-assessment with "Mock Tests".

4.4.3 View Assessments History

Students can view their past assessments that are older than a day from current date and review their previous test answers. This will help them to prepare for next assessment. The ability of showing the assessment results after a day provides the flexibility for instructors and administrator to conduct assessments to a group of students simultaneously. This whole feature is implemented in "UserTestHistory.aspx" asp.net page. We have used asp.net grid views to show the summary of the responses.

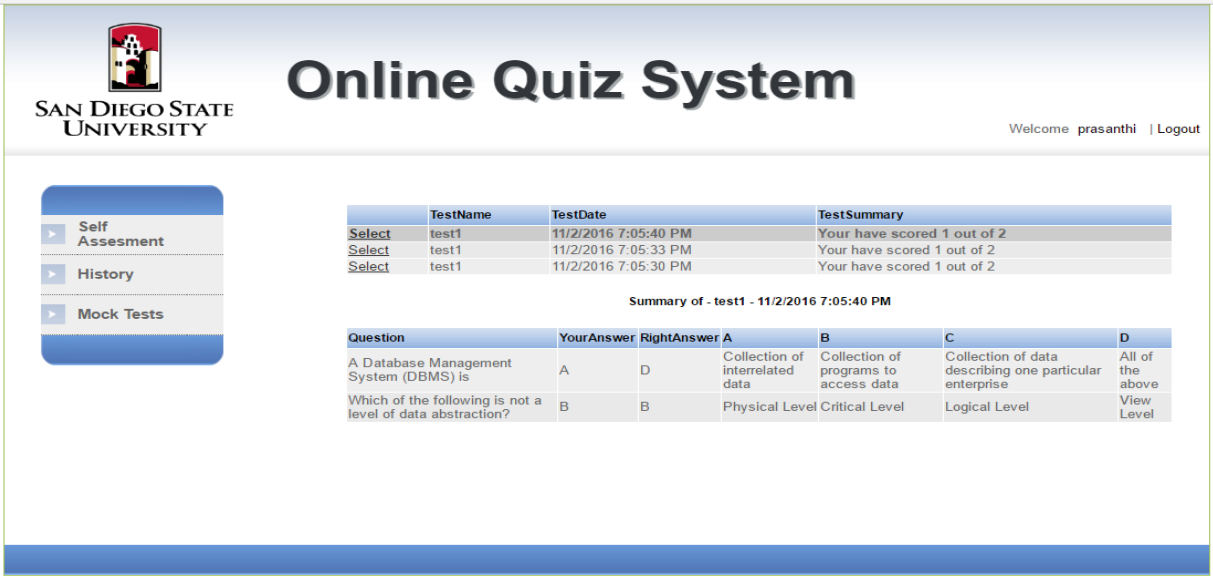

Figure 4.7. Assessment History.

4.5 ADMINISTRATOR DASHBOARD

When a user is authenticated into our system, there is an authorization check that verifies the role of the logged in user. If the logged in user has admin privileges, then he would be redirected to the admin dash board page.

The Admin is a super user of this application with highest privileges. Below are the allowed operations for this role.

1. Manage users roles and access permissions
2. Create new assessments
3. Create assessment reports/summary
4. Maintain source with latest questions and topics

Admin master page is named "Admin/**OnlineQuiz.Master**". This master page contains the layout markup common to all admin pages, the menu of admin actions is highlighted as shown in Figure 4.8.

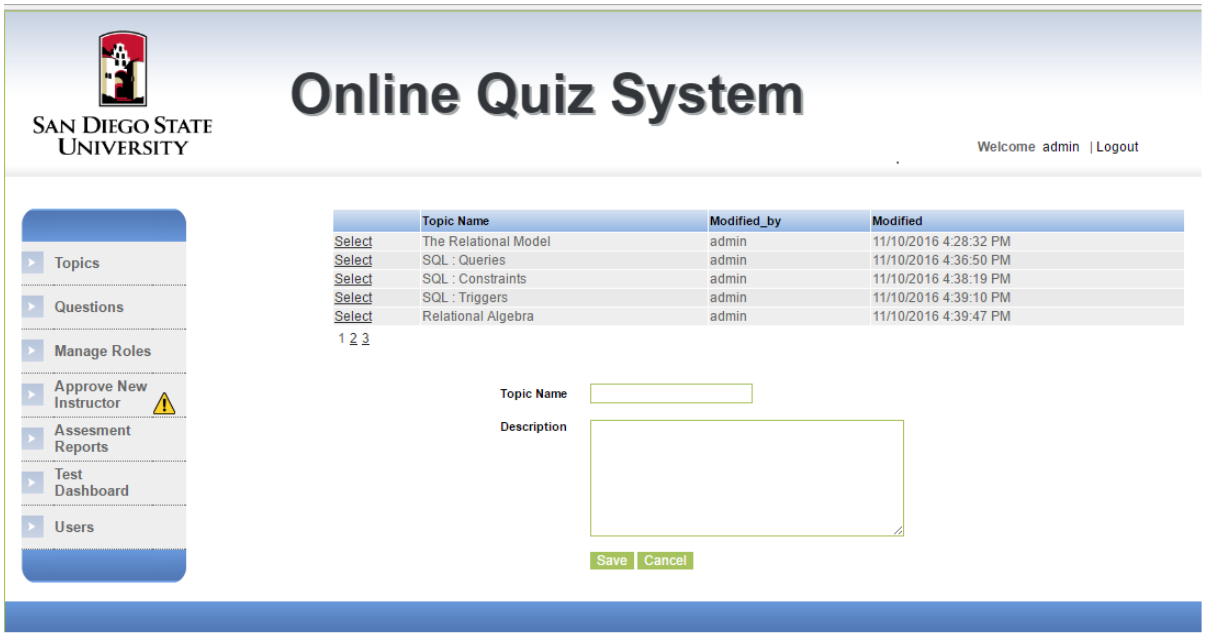

Figure 4.8. Admin page layout.

4.5.1 Manage User Roles

When the admin user clicks on “Manage User Roles”, the hyper link on the left action menu, he would be redirected to the “ManageRoles1.aspx” asp.net page. This web page shows the list of all enabled and disabled users. And admin can enable/disable a specific user in a few mouse clicks.

To show a user-friendly view, we have leveraged the asp.net “LIST BOX” concept to show the list of users in a box kind of view with images, showing “<<” and “>>”.Clicking on these arrows will move the user from the enabled list box to a disabled list box and vice versa. Also, to reduce the number of calls to server side code we relied on java script implementation. Clicking of left and right arrows (“<<” and “>>”) will not invoke server side code, instead we invoke JavaScript code and prepare the final enabled and disabled users list. Clicking of the final submit will save this final list into the database

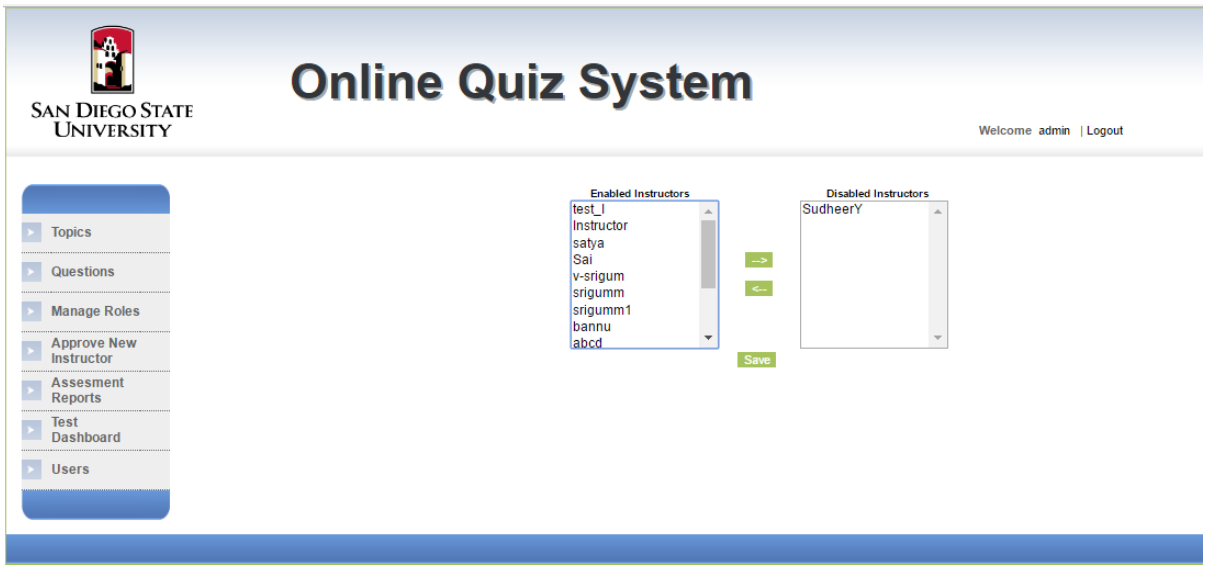

Figure 4.9. Manage User Roles.

4.5.2 Approve New Instructor

This is implemented as part of the "ManageRole.aspx" asp.net page. Here we have used "Asp.Net Gridview" to show the list of all new registered instructors, with checkboxes next to each instructor name. Admins can approve new instructors with a few mouse clicks and there is an option to approve new instructors as a batch as well.

4.5.3 Assessment Reports

Admins can view the summary of each assessment. This helps them to understand the overall performance of students for the specific assessment. Based on these results, admins can propose changes to the future assessments which can improve the whole performance assessment system of the university.

This has been implemented in "AssessmentReport.aspx" asp.net page. We used "ASP.NET GRID VIEW" to show the list of all assessments with a select button next to each assessment name. Clicking of this select button will hit the asp.net code, which in turn will query the database to get the summary data of this specific assessment.

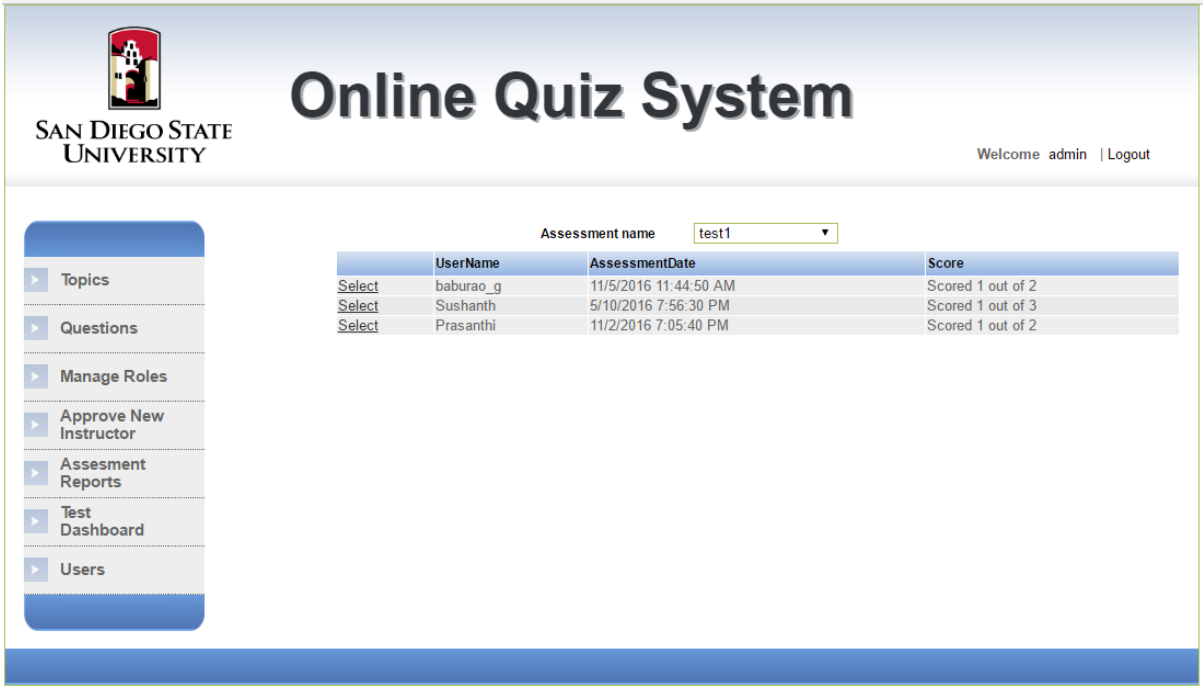

Figure 4.10. Assessment Reports.

4.5.4 Manage Assessment and Topics

Admin user has provision to load the system with new questions and keep it updated. To make this process easy, we have added a new page "OnlineQuizQuestions.aspx". This web page helps admins to maintain the question bank. Below are the major supported operations

1. Add new questions
2. Update existing questions
3. Delete outdated questions.

The questions are grouped together based on Topics. Online Quiz System consists of the following topics and has the provision to create new topics and add new questions.

1. The Relational Model.
2. SQL: Queries.
3. SQL: Constraints.
4. SQL: Triggers.
5. SQL: Nested Queries.
6. Relational Algebra.

7. Database Systems Design and Implementation.
8. Normalization.
9. ORACLE –JDBC.
10. Storage and File Structures.
11. Indexing and Hashing.
12. Query Processing Techniques.

Administrator has the provision to create and modify assessments by selecting topics for the assessments as shown in Figure 4.11.

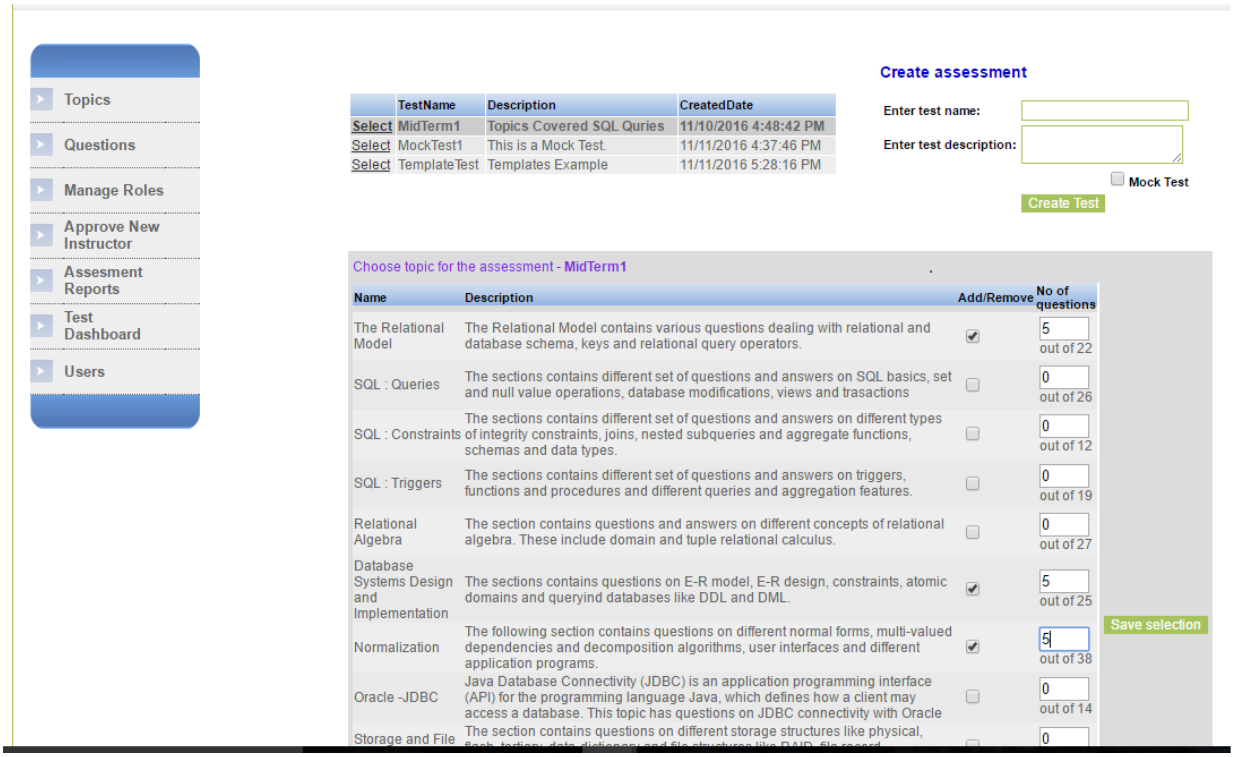

Figure 4.11. Manage Assessment and Topics.

4.6 INSTRUCTOR DASHBOARD

The instructor dashboard web page will be shown if the logged in user has **"Course Instructor"** privileges. Course Instructor Role is a subset of Admin role and has the following functionalities.

1. Create new questions
2. Create new topics
3. Create and view assessment reports/summary

The instructor dash board page layout is designed using the “Asp.Net master page” concept. Instructor actions are rendered as a menu of hyperlinks on the left side of the page; each hyperlink action click will redirect the user to the relevant page where he can perform the action related to that module.

While designing the instructor web pages, we have reused admin web pages to the best possible extent, as the privileges of an instructor are a subset of the admin user. These web pages are intelligent enough to change their layout structure based on the logged in user. To illustrate this with an example, we have “Admin/Site.master” asp.net page which is designed for the layout of admin pages, however we have reused this even for instructor dash board. This web page shows only three action links on the left menu for Instructor roles, while showing five actions for admin roles. Same web page can show different html view based on the logged in user.

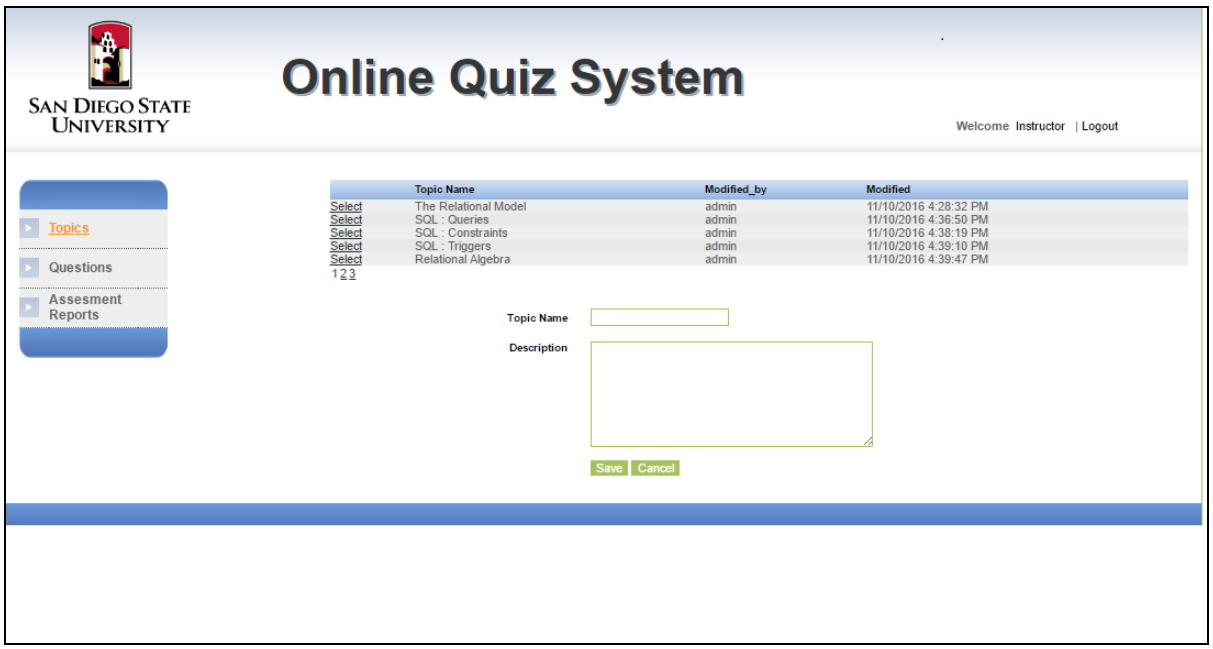

Figure 4.12. Admin master page reused for Instructor.

4.6.1 Assessment Reports

Instructors can view the summary of each assessment. This helps them to understand the overall performance of students for the specific assessment, and based on these results,

faculty can re-examine the methodologies and focus on the required learning areas for students.

We have reused the existing Asp.net web page ("AssessmentReport.aspx") in our system, which is designed for the admin role. Since the report creation part is the same for both user types we have reused the same web page without creating new web page.

4.6.2 Manage Topics and Questions

This feature enables instructors to manage topics and questions (add/edit/delete). With this feature we can always keep this application up to date with the latest syllabus.

We have reused the "**OnlineQuizQuestions.aspx**" and "**OnlineQuizTopics.aspx**" asp.net pages to implement this functionality.

4.7 DATABASE IMPLEMENTATION

This web application is completely data driven, and there is nothing hard coded in the code files or static files like xml and text files. Everything that we see on the user page is out of a database.

Below is the list of database objects:

4.7.1 Tables

Answer Table: It contains records about the user selected answers. This table is used generate assessment history. The Question_Id column in this table references the Question_Id column in Questions Table. This table is used by the following stored procedures.

- OQS_GetAllUsersTestSummary
- OQS_GetUserTestHistory
- OQS_GetUserTestResponses
- OQS_InsertAnswer
- OQS_SaveUserQuestionAnswer

Table 4.1. Answer

[dbo].[OQS_Answer]

Columns

Key	Name	Data Type	Max Length (Bytes)	Allow Nulls
FK	Question_Id	int	4	False
	Answer	varchar(50)	50	True
	UserID	int	4	True
	Test_Id	int	4	True
	Created_By	varchar(100)	100	True
	Created_Date	datetime	8	True
	Modified_By	varchar(100)	100	True
	Modified_Date	datetime	8	True
	Is_Deleted	bit	1	True
	setno	int	4	True

Foreign Keys

Name	Columns
FK_OQS_Answer_OQS_Questions	Question_Id->[dbo].[OQS_Questions].[Question_Id]

Question Table: This table holds the list of question in The Online Quiz system. As the questions are grouped into topics, this table consists of topicId that references topicId of Topic Table.

The following stored procedures uses this table.

- OQS_Answer
- OQS_CreateQuestion
- OQS_DeleteQuestion
- OQS_GetAllQuestions
- OQS_GetQuestionsInfo_ByTopicId
- OQS_GetTestQuestions
- OQS_GetUserTestResponses
- OQS_UpdateQuestion

Table 4.2. Questions

[dbo].[OQS_Questions]						
Key	Name	Data Type	Max Length (Bytes)	Allow Nulls	Identity	Default
FK C	Question_Id	int	4	False	1 - 1	
FK	Topic_Id	int	4	True		
	Question_Name	varchar(500)	500	False		
	Option1	varchar(150)	150	True		
	Option2	varchar(150)	150	True		
	Option3	varchar(150)	150	True		
	Option4	varchar(150)	150	True		
	Answer	varchar(500)	500	True		
	Created_By	varchar(50)	50	True		
	Created_Date	datetime	8	True		(getdate())
	Modified_By	varchar(50)	50	True		
	Modified_Date	datetime	8	True		
	Is_Deleted	bit	1	True		((0))
	IsTemplate-Question	bit	1	True		

Foreign Keys

Name	Columns
FK_OQS_Questions_OQS_Topics	Topic_Id->[dbo].[OQS_Topics].[Topic_Id]

Role Table: This table hold the information about the user roles in Online Quiz System Application.

It is used by the following stored procedures.

- OQS_CreateRole
- OQS_DeleteRole
- OQS_GetAllRoles
- OQS_GetAllUsers
- OQS_LoginQuiz

Table 4.3. Role

[dbo].[OQS_ROLE]

Columns

Key	Name	Data Type	Max Length (Bytes)	Allow Nulls	Identity
PK	ROLE_ID	int	4	False	1 - 1
	ROLE_NAME	varchar(50)	50	True	
	ISACTIVE	bit	1	True	
	CREATED_BY	varchar(50)	50	True	
	CREATED_DATE	varchar(50)	50	True	
	MODIFIED_BY	varchar(50)	50	True	
	MODIFIED_DATE	varchar(50)	50	True	
	ROLE_DESCRIPTION	varchar(500)	500	True	

Test Table: This table contains list of assessments in the Online Quiz System Application. It is used by the following stored procedures.

- OQS_TestTopics
- OQS_CreateTest
- OQS_GetAllActiveTests
- OQS_GetAllTests
- OQS_GetUserTestHistory

Table 4.4. Test

[dbo].[OQS_Test]

Columns

Key	Name	Data Type	Max Length (Bytes)	Allow Nulls	Identity	Default
PK	Id	int	4	False	1 - 1	
	Name	varchar(250)	250	False		
	Description	ntext	max	True		
	CreatedBy	varchar(50)	50	False		
	CreatedDate	datetime	8	False		
	ModifiedBy	varchar(50)	50	True		
	ModifiedDate	datetime	8	True		
	IsEnabled	bit	1	False		((1))

Test Topics Table: This is a mapping table between Assessments and topics. TestId filed in this table references Id of Test table and TopicId references Topic_Id of topic table. The following stored procedures uses this table.

- OQS_AddTopicToTest
- OQS_GetTestQuestions
- OQS_GetTestTopics
- OQS_RemoveTopicFromTest

Table 4.5. Test Topics

[dbo].[OQS_TestTopics]

Columns

Key	Name	Data Type	Max Length (Bytes)	Allow Nulls
FK	TestId	int	4	True
FK	TopicId	int	4	True

Foreign Keys

Name	Columns
FK_OQS_TestTopics_OQS_Test	TestId->[dbo].[OQS_Test].[Id]
FK_OQS_TestTopics_OQS_Topic	TopicId->[dbo].[OQS_Topics].[Topic_Id]

Topics Table: Topics Table holds information about list of topics available in the OQS application. It is used by the following stored procedures

- OQS_Questions
- OQS_TestTopics
- OQS_DeleteTopic
- OQS_GetAllTopics
- OQS_GetTestTopics
- OQS_GetTopicName
- OQS_InsertTopic
- OQS_UpdateTopicDescription

Table 4.6. Topics

[dbo].[OQS_Topics]

Columns

Key	Name	Data Type	Max Length (Bytes)	Allow Nulls	Identity
PK C	Topic_Id	int	4	False	1 - 1
	Topic_Name	varchar(250)	250	False	
	Topic_Description	ntext	max	True	
	Views	int	4	True	
	Created_By	varchar(50)	50	False	
	Created_Date	datetime	8	False	
	Modified_By	varchar(50)	50	True	
	Modified_Date	datetime	8	True	
	Is_Deleted	bit	1	False	

User Information Table: All the user information for the Online Quiz System is stored in this table. It is used by the following stored procedures.

- OQS_CreateUser
- OQS_EnableInstructor
- OQS_GetAllInstructors
- OQS_GetAllUsers
- OQS_GetAllUsers¬Test¬Summary
- OQS_GetInstructors
- OQS_GetNewInstructors
- OQS_LoginQuiz

Table 4.7. User Information

[dbo].[OQS_UserInformation]

Columns

Key	Name	Data Type	Max Length (Bytes)	Allow Nulls	Identity	Identity Replication
PK C	UserID	int	4	False	1 - 1	False
	Username	varchar(30)	30	False		
	Password	varchar(30)	30	False		
	IsActive	bit	1	True		
	Role	int	4	True		
	Enable-Access	bit	1	True		
	Firstname	varchar(50)	50	True		
	Lastname	varchar(50)	50	True		
	Email_Id	varchar(50)	50	True		
	Created_By	varchar(50)	50	True		
	Created_Date	varchar(50)	50	True		
	Modified_-Date	varchar(50)	50	True		
	Modified_By	varchar(50)	50	True		

TemplateQuestionOptionsInfo Table: This table is used to hold answer and the available multiple choice options to the template questions. The following stored procedures uses this table.

- OQS_GetTestQuestions
- OQS_GetUserTestResponses

Table 4.8. TemplateQuestionOptionsInfo

[dbo].[TemplateQuestionOptionsInfo]

Columns

Name	Data Type	Max Length (Bytes)	Allow Nulls
QuestionId	int	4	True
SetNo	int	4	True
Option1	nvarchar(100)	200	True
Option2	nvarchar(100)	200	True
Option3	nvarchar(100)	200	True
Option4	nvarchar(100)	200	True
Answer	nvarchar(100)	200	True

TemplateQuestionSetInfo Table: This table is used to hold data about various set of values for each templated question. The questionId in this table references question Id of the Question table. This table is used by the following stored procedures.

- OQS_GetTestQuestions
- OQS_GetUserTestResponses

Table 4.9. TemplateQuestionSetInfo

[dbo].[TemplateQuestionSetInfo]

Columns

Name	Data Type	Max Length (Bytes)	Allow Nulls
QuestionId	int	4	True
SetNo	int	4	True
PlaceholderPosition	int	4	True
value	nvarchar(200)	400	True

CHAPTER 5

SUMMARY AND OBSTACLES

Working on this project was an interesting, challenging, intellectually enriching experience and taught me the importance of time management, data collection, design and planning. It not only gave me an opportunity to work with latest technology and enhance my skills but also taught me to focus on user interface and design decisions.

Some of the challenging tasks of the thesis were the collection of questions about Relational Database Management System Concepts, framework support for templated questions and ability for Admin and instructor to a select list of topics for an assessment.

A major obstacle was to support different browsers. Most of the CSS properties and HTML5 properties did not work very well with Internet Explorer and certain features were not compatible with Chrome and Firefox, so many workarounds and validations had to be coded to ensure the application worked on all browsers. However, it was determined that Firefox was best suited for it and was made a requirement for the application.

Professor Eckberg's support and suggestions played a major role in overcoming most of the obstacles and making this application successful.

CHAPTER 6

FUTURE ENHANCEMENTS

This tool provides a framework, which can be expanded to add more features and functionalities as discussed below.

6.1 EMAIL NOTIFICATIONS

The Online Quiz System application maintains a database of student information and their assessment reports. The application can be enhanced so that the Admin and faculty team has the provision to auto deliver assessment reports to students though email. They can also broadcast to students when additional assessments and mock tests are created on the portal.

6.2 DATA MINING

The Online Quiz System holds information about student's assessments data. The system can be enhanced to data mine students answers and create reports about problem areas that students are facing. This report can be helpful for Admin and faculty team to determine difficult topics and helps them to concentrate on those particular topics. They can also create additional study material about that topic in order to help students.

6.3 MORE TEMPLATED QUESTIONS

We have provided the framework to support templated questions. We can enhance the application by adding more templated questions to the database. This helps students to practice concepts using mock tests and faculty can conduct quizzes without worrying about an insufficient number of questions in the database.

REFERENCES

[1] WIKIPEDIA, *HTML*. Wikipedia, https://en.wikipedia.org/wiki/HTML, accessed October 2016, n.d.

[2] EVINCE TECHNOLOGIES, *Pros & cons of HTML5*. Evince Technologies, https://evincetech.wordpress.com/2013/02/02/pros-cons-of-html5/, accessed November 2016, 2013.

[3] WIKIPEDIA, *CSS*. Wikipedia, https://en.wikipedia.org/wiki/Cascading_Style_Sheets, accessed October 2016, n.d.

[4] SANFORD-BROWN, *The pros and cons of CSS*. Sanford-Brown, http://www.sanfordbrown.edu/Student-Life/blog/January-2015/Pros-and-Cons-of-CSS, accessed November 2016, 2015.

[5] TUTORIALS POINT, *JavaScript overview*. Tutorials Point, http://www.tutorialspoint.com/javascript/javascript_overview.htm, accessed November 2016, n.d.

[6] D. MARSHALL, *Pros and cons of JavaScript*. Cardiff School of Computer Science & Informatics, http://users.cs.cf.ac.uk/Dave.Marshall/Internet/node287.html, accessed November 2016, 2001.

[7] WIKIPEDIA, *C-Sharp (programming language)*. Wikipedia, https://en.wikipedia.org/wiki/C_Sharp_(programming_language), accessed November 2016, n.d.

[8] DREAM.IN.CODE, *Pros/cons of the C# language*. Dream.In.Code, http://www.dreamincode.net/forums/topic/29142-proscons-of-the-c%23-language/, accessed November 2016, 2007.

[9] WIKIPEDIA, *ASP.NET*. Wikipedia, https://en.wikipedia.org/wiki/ASP.NET, accessed November 2016, n.d.

[10] STACK OVERFLOW, *When to choose ASP.NET and when PHP?* Stack Overflow, http://stackoverflow.com/questions/3368073/when-to-choose-asp-net-and-whenphp, accessed November 2016, 2010.

[11] ORIENT SOFTWARE, *ASP.NET vs PHP*. Orient Software, http://www.orientsoftware.net/technologies/microsoft-net/aspnet-vs-php, accessed November 2016, n.d.

[12] WIKIPEDIA, *Microsoft SQL server*. Wikipedia, https://en.wikipedia.org/wiki/Microsoft_SQL_Server, accessed November 2016, n.d.

[13] M. LI, *Templated Q&A DB system for independent study and self-assessment –Oracle back end design and implementation*, Master's thesis, San Diego State University, San Diego, CA, 2002.

[14] H. KRAMER, *Templated Q&A DB system for independent study and self-assessment – Java front end design and implementation*, Master's thesis, San Diego State University, San Diego, CA, 2002.